Exploring Audio Features and Genre Classification

Introduction

In the vast expanse of modern music consumption, with new melodies and rhythms continuously being streamed across different digital platforms, the task of organizing and classifying this vast audio landscape poses an immense challenge. As music tracks proliferate on albums, streaming services, and beyond, the distinctions between genres blur into ambiguity, making the classification of songs into distinct categories a formidable challenge.

At the forefront of efforts to tame this musical chaos stands automatic genre classification—a field that has long captivated researchers in the domain of Music Information Retrieval (MIR). For decades, these pioneers have sought innovative techniques to unravel the complex tapestry of musical diversity, sorting music into discernible genres and styles.

The exploration of audio features and genre classification has been a topic of significant interest in the field of music information retrieval. This project aims to delve into the intricacies of this subject, focusing on the extraction of audio features and the subsequent classification of music genres.

The importance of genre classification is underscored by the work of Cory and Ichiro, who poses the question of whether pursuing musical genre classification is worthwhile. They argue that genre classification can be improved by focusing on specific characteristics of music, thereby highlighting the need for a more nuanced approach to this task.

Our project also draws inspiration from the research conducted by Li et al., who have made significant strides in the field of audio feature extraction. Their work emphasizes the importance of extracting robust and discriminative features from audio signals for effective genre classification.

Furthermore, we consider the work of Panagakis et al. to be instrumental in shaping our understanding of genre classification. Their research provides a comprehensive overview of various audio feature extraction techniques and their effectiveness in genre classification tasks.

Considering these sources, our project seeks to explore audio features and genre classification in a comprehensive manner, with the goal of contributing to the ongoing discourse in this fascinating field of study.

In this paper we are using Spotify’s rich array of audio features - such as acousticness, liveness, speechiness, instrumentalness, energy, loudness, danceability, valence, duration, tempo, key, and mode, we endeavor to uncover patterns and characteristics indicative of different musical genres. We are using diverse samples of songs spanning six broad genres: pop, rap, rock, Latin, EDM, and R&B. Through the lens of decision trees, random forests, and XGBoost models, we navigate the melodic landscape, seeking to discern the subtle nuances that define each genre.

Methodology

Dataset Background

The dataset used for this project was found on kaggle.com:
https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset

Data Exploration

The dataset imported containing audio features of 114,000 songs from Spotify. The dataset consisted of 21 columns including track ID, artist, album name, track name, popularity, duration, explicit content, and various audio features such as danceability, energy, loudness, and tempo. Initial exploration involved summarizing the dataset and visualizing the distribution of audio features across different genres using density plots to understand the distribution of songs across different genres.

Genre Selection and Data Filtering

For focused analysis, we selected six genres of interest: classical, rock, R&B, pop, Latin, and EDM. We filtered the dataset to include only those entries where the **track_genre** matches one of our selected genres. This results in a filtered dataset, **filtered_songs**, containing 6,000 entries, with 1,000 entries per genre. This filtered dataset served as the basis for subsequent analyses.

Feature Analysis and Visualization

We identified 12 audio features including duration, danceability, energy, key, mode, speechiness, acousticness, instrumentalness, liveness, valence, and tempo. Visualization techniques such as density plots were employed to understand the distribution and relationships between these features within and across genres *(Figure1)*. These features provide a comprehensive representation of the audio characteristics of a song.

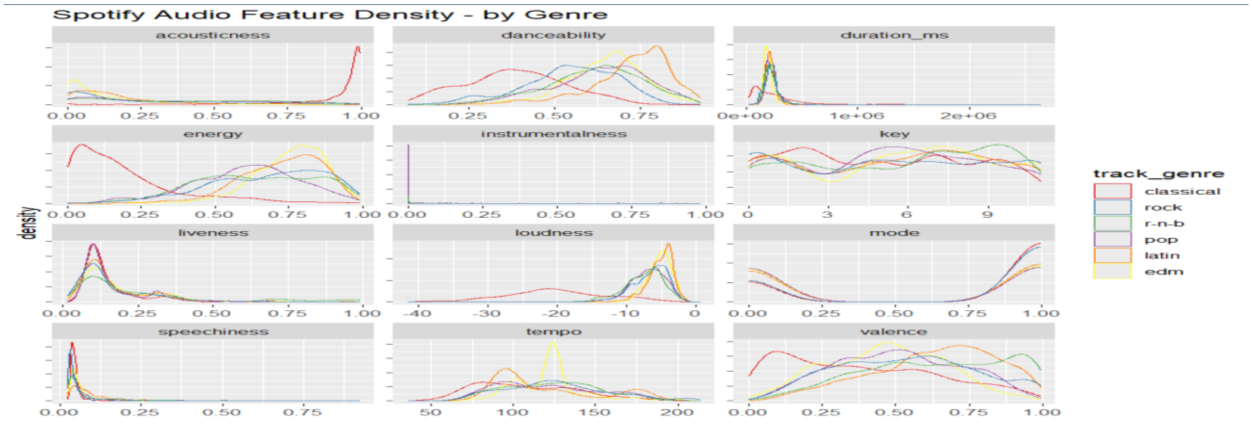

Figure 1: Audio Feature Density by Genre

Outlier Detection and Removal

In any data analysis task, outliers can significantly skew the results and lead to inaccurate conclusions. In our dataset, we noticed that the duration_ms feature contained some extreme values that could potentially affect our analysis.

To identify these outliers, we used the boxplot function, which visualizes the distribution of a numerical variable and highlights any values that fall outside a given range. After identifying the outliers, we removed them from our dataset. This was done by filtering out any songs whose duration_ms value was identified as an outlier.

To visualize the effect of this outlier removal, we created two boxplots: one before and one after the outlier removal *(Figure2)*.

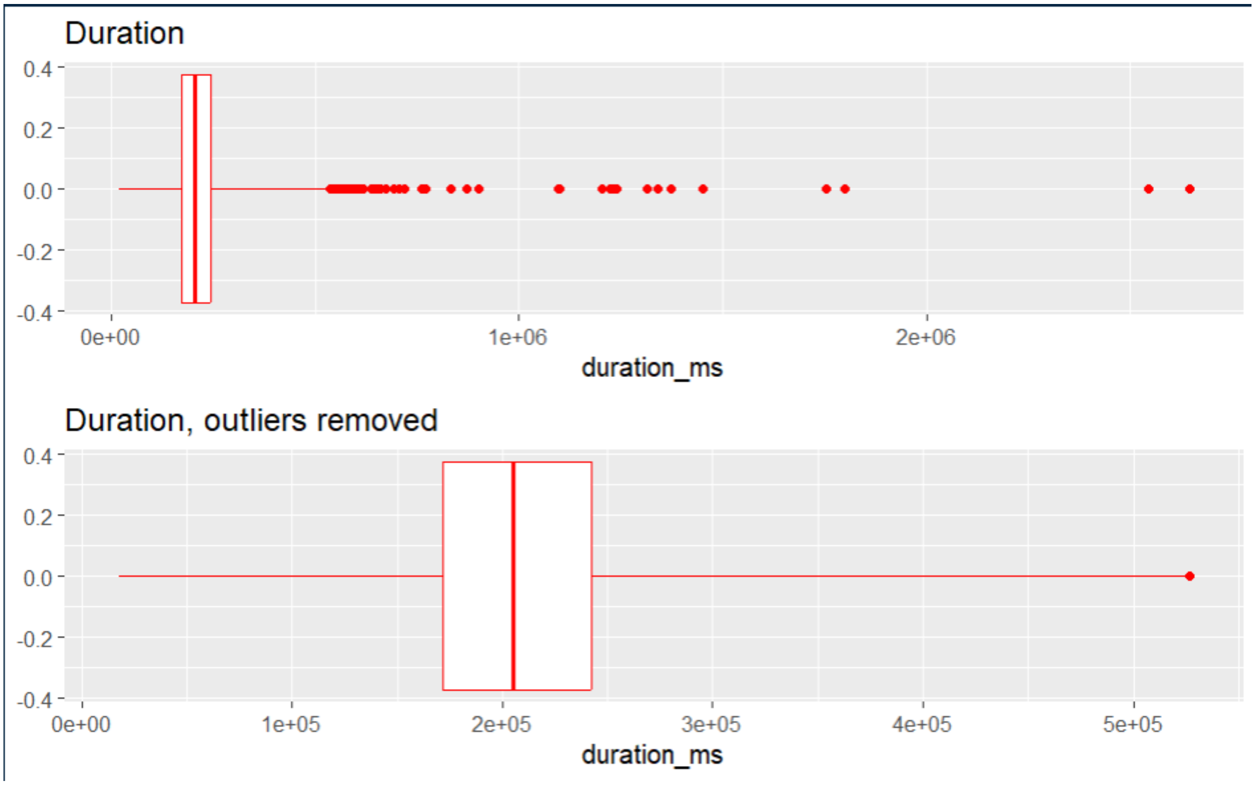

Figure 2: Duration (with and without outliers)

Correlation Analysis

We examined the correlation between audio features to identify redundant or highly correlated features. A correlation matrix was generated and visualized to illustrate the relationships between features *(Figure3(Left))*. Notable correlations included the strong positive correlation between energy and loudness, and the negative correlation between energy and acousticness. This analysis helps us understand the relationships between different features and their potential impact on genre classification.

Genre Correlation Analysis

Median feature values were calculated for each genre, and correlations between genres were explored based on these median values *(Figure3(Right))*. Classical music showed negative correlations with all other genres, while EDM and R&B exhibited positive correlation. This analysis provides insights into which genres are most similar or dissimilar based on their audio features.

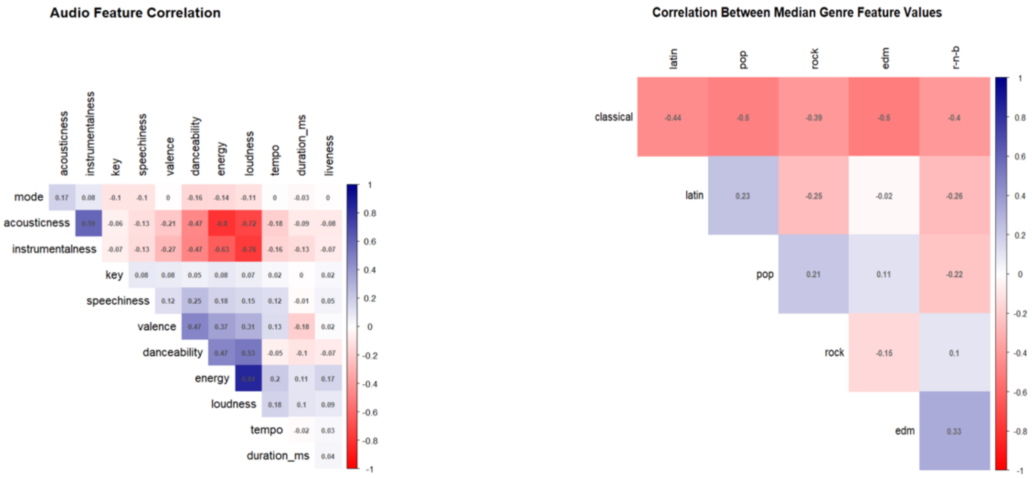

Figure 3: Audio Feature Correlation (Left) Median Genre Feature Values Correlation (Right)

Genre Classification

Data Preparation

The data preparation phase involved scaling the numeric features and splitting the dataset into a training set (80% of the songs) and a test set (20%). The steps involved in this process were as follows:

Scaling Numeric Features: All numeric features in the dataset were scaled using the scale() function to standardize them and bring them to a common scale, ensuring that no single feature dominates the modeling process due to its larger magnitude.

Splitting the Dataset: The scaled dataset was then split into a training set and a test set using a 80-20 split ratio. This was achieved by randomly sampling 80% of the rows to form the training set while the remaining 20% formed the test set. The sample() function with replace = FALSE ensured that each observation was selected only once for either the training or test set.

Modeling

A variety of classification models were considered to classify genres based on audio features. Classification models that were considered include Decision Tree, Random Forest, and Gradient

Boosting with XGBoost. Each model underwent evaluation to determine its accuracy in categorizing songs into their respective genres.

Model Evaluation

We assess the performance of each classification model by calculating overall accuracy and examining confusion matrices. This allows us to compare the effectiveness of different models in classifying songs into genres.

Variable Importance Analysis

We analyze the importance of different audio features in genre classification using each model's feature importance metrics. This analysis helps us identify which features are most influential in determining the genre of a song.

Results

Decision Tree

We first performed decision tree model. The decision tree model was constructed using the rpart package in R. Decision trees are simple classification tool, where each node represents a feature, each branch an outcome of a decision on that feature, and the leaves represent the class of the final decision. The algorithm partitions the data into sub-spaces repeatedly to create the most homogeneous groups possible, generating rules that are visualized in the tree structure.

The decision tree model was trained on the training dataset and visualized using the rpart.plot function*(Figure5)*. Interpretation of the decision tree revealed the most important feature ('acousticness'), which separates classical from the rest of the classes on the first decision. The model's performance was evaluated on the test dataset, where it achieved an overall accuracy of 52%(*Table1*).

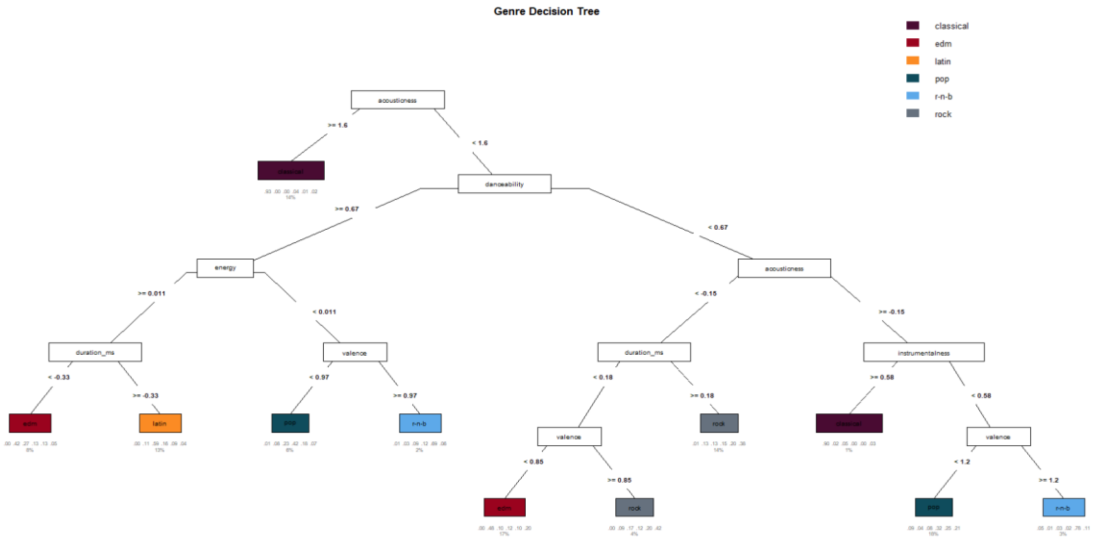

Figure 4: Genre Decision Tree

Table 1: Genre Decision Tree Accuracy

Match	N	Accuracy	Model
FALSE	576	0.4848485	decision_tree
TRUE	612	0.5151515	decision_tree

Random Forest

The random forest model was constructed using the randomForest function in R, creating an ensemble of 100 decision trees (ntree = 100). Random forests leverage a technique called bootstrap aggregating or bagging, where multiple decision trees of varying depths are trained on different subsets of the data, and their classifications are aggregated.

This ensemble approach helps mitigate overfitting and enhances prediction accuracy compared to a single decision tree model. After training the random forest model, its predictive performance was evaluated on the test dataset using the predict function. The random forest model achieved an overall accuracy of 70% on the test set, correctly predicting the genres for approximately 70% of the songs (*Table2*). By employing the random forest algorithm, the study harnessed the power of ensemble learning to improve the robustness and accuracy of song genre prediction.

Table 2:Genre Random Forest Accuracy

Match	N	Accuracy	model
FALSE	362	0.3047138	random_forest
TRUE	826	0.6952862	random_forest

Gradient boosting with XGBoost

The next round of improvements to the random forest model come from boosting, or building models sequentially, minimizing errors and boosting the influence of the most successful models. Adding in the gradient descent algorithm for minimizing errors results in a gradient boosting model. Here, we have used XGBoost, which provides parallel processing to decrease compute time as well as various other improvements.

We have used the XGBoost function with most of the default hyperparameter settings, just setting objective to handle multiclass classification.

The implementation of gradient boosting with XGBoost for genre classification involved preparing the training and test dataset by converting it into a matrix format compatible with XGBoost. The XGBoost model was then trained on the training data, specifying parameters such as the number of boosting rounds and the objective function for multiclass classification. After training, the model was used to make predictions on the test data, generating genre labels for the songs.

The accuracy assessment for the XGBoost model reveals that out of a total of 1188 predictions, 362 were incorrectly classified and 826 predictions were correctly classified, yielding an accuracy of 70% (*Table3*).

Table 3:Genre Gradient boosting with XGBoost Accuracy

Match	N	Accuracy	Model
FALSE	362	0.3047	xgboost
TRUE	826	0.6953	xgboost

Discussion

Model Comparisons:

Variable Importance Measures:

Despite using different metrics, it's common to find that certain features consistently rank highly across all models. These features typically serve as the primary differentiators in separating songs into distinct genres, as they lead to substantial reductions in impurity or improvements in accuracy.

For decision trees the most important variable is typically the one that appears at the root node of the tree, here it is acousticness, followed by danceability, energy, duration, and instrumentalness. The initial split results in the largest reduction in impurity.

In case of Random Forest, importance of variables is measured by the mean decrease in node impurity. Features with higher mean decrease values are considered more important. Here, acousticness tends to have the highest importance, followed by danceability, energy, duration, and instrumentalness.

XGBoost model variable importance is determined by the gain achieved by including a feature in the model. Features with higher gains contribute more to the improvement in accuracy. Acousticness is again the most important variable, followed by danceability, energy, duration, and instrumentalness *(Figure6)*.

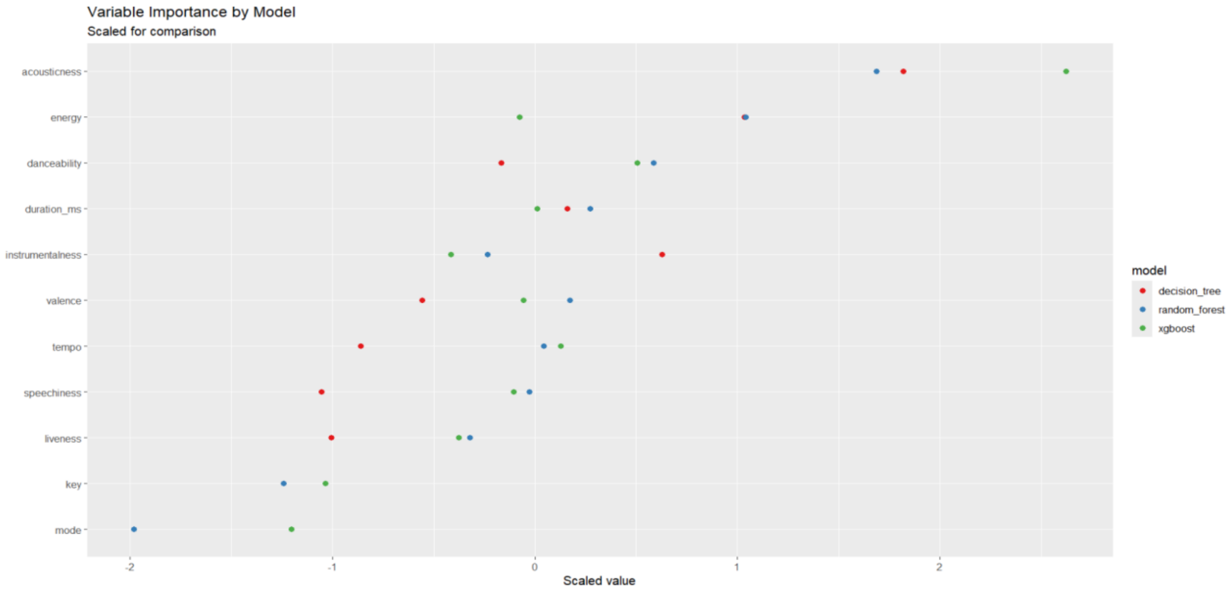

Figure 5: Variable Importance by Model

Overall, across all three models, acousticness consistently emerges as the most important variable for predicting genres. Other important variables include danceability, energy, duration, and instrumentalness. These variables play crucial roles in distinguishing between different music genres.

Accuracy:

Both the Random Forest and XGBoost models performed similarly well, with an accuracy of 70%. However, the Decision Tree model had a lower accuracy of 52%. The lower accuracy of the Decision Tree model compared to the Random Forest and XGBoost models could be attributed to its inherent limitations. Decision trees are prone to overfitting, especially when dealing with complex datasets. They tend to create overly complex models that capture noise in the data, leading to poorer generalization performance on unseen data.

Table 4: Model Accuracy Comparison

Model	**Accuracy**
Random Forest	70%
Decision Tree	52%
XGBoost	70%

Genre Accuracy by model:

All genres except classical showed gains in accuracy as we moved from simpler decision tree to more complex models like random forest and XGBoost. Classical music proved to be the most challenging genre, with low accuracy across all models. This could be because classical music often lacks distinct features that differentiate it from other genres, especially in terms of audio features commonly used for classification. Unlike genres like EDM or rock, classical music may not exhibit clear-cut patterns or signatures that can be easily recognized by machine learning algorithms.

The random forest and XGBoost models demonstrated relatively high accuracy (73%) in classifying rock songs, indicating their effectiveness for this genre. However, all models struggled with accurately identifying R&B and pop songs, with accuracies ranging from 16% to 55.9% and 43% to 50.3%, respectively. This suggests that these genres may share similarities with others, making them harder to distinguish. For Latin music, the random forest model performed the best (69% accuracy), closely followed by XGBoost (68%), while the decision tree model also performed decently (44% accuracy). EDM songs were classified with relatively high accuracy across all models (69% to 78%), potentially due to distinct features that make them easier to identify.

The decision tree model struggled the most with accuracy across all genres, while the random forest and XGBoost models generally performed better, especially for genres like rock and EDM.

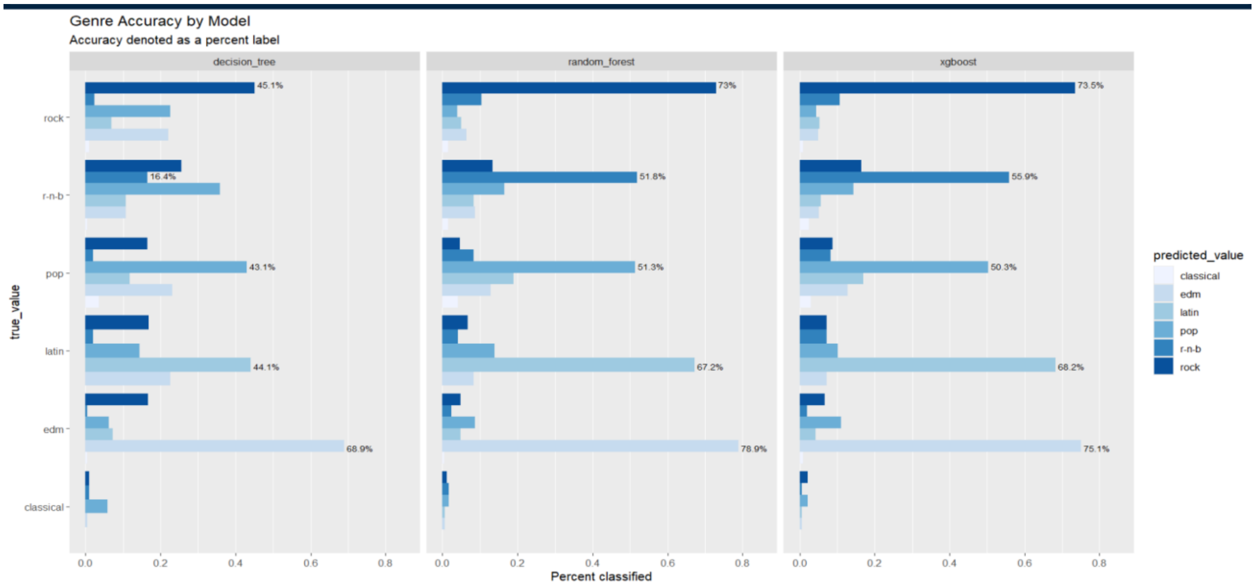

Figure 6: Genre Accuracy by Model

Limitations:

Representation could affect the generalizability of the findings and the performance of the classification models, particularly if there are significant variations within each genre or if certain genres are underrepresented in the dataset, the model's performance may not accurately reflect real-world scenarios.

Some genres share common audio features, making it challenging for the model to accurately classify songs into specific genres. This challenge reflects the inherent ambiguity and overlap between certain music genres, where songs can exhibit characteristics of multiple genres simultaneously. Manual genre classification by human experts may also face similar difficulties due to subjective interpretations of genre boundaries.

While the study utilizes a rich array of audio features provided by Spotify, there may be other relevant features that were not included in the analysis.

References:

Li, T., Ogihara, M., & Li, Q. (2003). A comparative study on content-based music genre classification. Retrieved from A-Comparative-Study-on-Content-Based-Music-Genre-Classification.pdf (researchgate.net)

Panagakis, Y., Kotropoulos, C., & Arce, G. R. (2009). Music genre classification via topology preserving non-negative tensor factorization and sparse representations. Retrieved from Music genre classification via Topology Preserving Non-Negative Tensor Factorization and sparse representations | IEEE Conference Publication | IEEE Xplore

Cory McKay and Ichiro Fujinaga (2006). Musical genre classification: Is it worth pursuing and how can it be improved? Retrieved from Microsoft Word - McKay2006Musical_cr_02.doc (mcgill.ca)